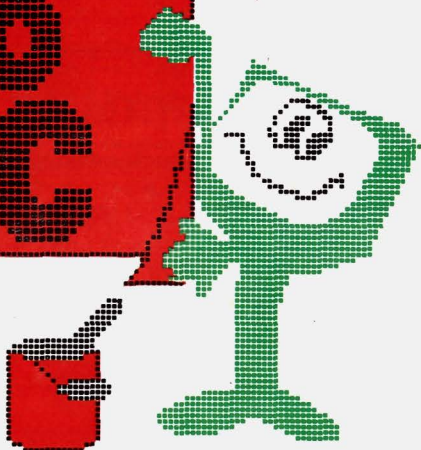
20 VIDEOLEZIONI DI BASIC
PER IMPARARE COL VIC 20

GRUPPO
EDITORIALE
JACKSON

6

COMMODORE
VIC 20

VIDEO BASIC VIC 20
Pubblicazione quattordicinale
edita dal Gruppo Editoriale Jackson
Direttore Responsabile:
Giampietro Zanga
Direttore e Coordinatore
Editoriale: Roberto Pancaldi
Autore: Softidea - Via Indipendenza 88 - Como
Redazione software:
Francesco Franceschini, Enrico Braglia,
Fabio Calanca
Segretaria di Redazione:
Marta Menegardo
Progetto grafico:
Studio Nuovaidea - Via Longhi 16 - Milano
Impaginazione:
Silvana Corbelli
Illustrazioni:
Cinzia Ferrari, Silvano Scolari
Fotografie:
Marcello Longhini
Distribuzione: SODIP
Via Zuretti, 12 - Milano
Fotocomposizione: Lineacomp S.r.l.
Via Rosellini, 12 - Milano
Stampa: Grafika '78
Via Trieste, 20 - Pioltello (MI)
Direzione e Redazione:
Via Rosellini, 12 - 20124 Milano
Tel. 02/6880951/5

Autorizzazione alla pubblicazione Tribunale di Milano n° 422 del 22-9-1984
Spedizione in abbonamento postale Gruppo II/70 (autorizzazione della Direzione Provinciale delle PPTT di Milano).
Prezzo del fascicolo L. 8.000
Abbonamento comprensivo di 5 raccoglitori L. 165.000
I versamenti vanno indirizzati a: Gruppo Editoriale Jackson S.r.l. - Via Rosellini, 12 20124 Milano, mediante emissione di assegno bancario o cartolina vaglia oppure utilizzando il c.c.p. n° 11666203.
I numeri arretrati possono essere richiesti direttamente all'editore inviando L. 10.000 cdu. mediante assegno bancario o vaglia postale o francobolli.
Non vengono effettuate spedizioni contrassegno.

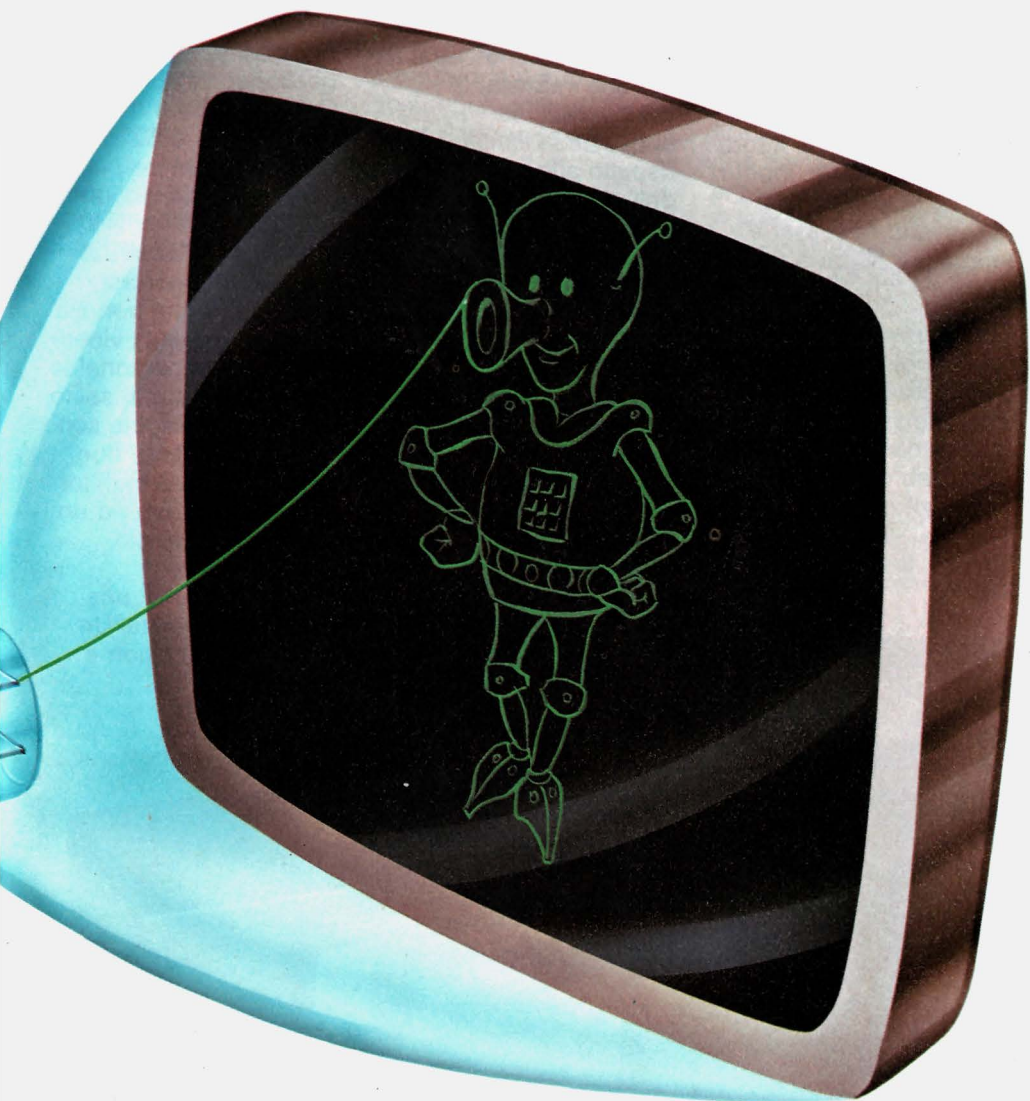
Gruppo Editoriale Jackson

SOMMARIO

Introduzione

Il tuo computer è, tra le altre cose, anche una speciale stazione emittente televisiva. È infatti in grado di trasmettere via cavo le informazioni elaborate. Sotto questo aspetto, il programmatore diventa il "regista" dell'output dei dati trattati.
A lui sono, cioè, demandate le responsabilità di dare alle informazioni in uscita il massimo risalto e la maggiore chiarezza possibile.
Da qui l'importanza di conoscere sia l'hardware dedicato alla visualizzazione delle immagini, che le istruzioni e funzioni del BASIC capaci di controllare il formato e gli attributi.
Molto del successo dei tuoi programmi futuri dipende dalla familiarità con questi elementi.

Televisori e monitor

Il televisore ed il monitor - o, più generalmente, le unità video - costituiscono il principale dispositivo di uscita di un computer. Ad esse, infatti, viene normalmente affidato il compito di visualizzare tutte le informazioni, i dati ed i messaggi che costituiscono la base del fondamentale rapporto di interazione tra l'uomo e l'elaboratore.
Le funzioni svolte da un'unità video sono fondamentalmente tre:
1) visualizzare sullo schermo la maggior parte dei caratteri battuti sulla tastiera (tale funzione è chiamata eco);
2) visualizzare l'output dei programmi, consentendo di risparmiare tempo e carta quando non è necessaria una copia permanente;
3) inviare all'utente i messaggi dell'interprete BASIC (ad esempio le segnalazioni di errore).
L'uso delle unità video come dispositivi di output è abbastanza recente: fino a pochi anni fa le informazioni in uscita dal calcolatore erano infatti svolte quasi esclusivamente da stampanti e telescriventi. Ben presto, però, ci si rese conto che tali

HARDWARE

dispositivi erano assolutamente insufficienti per far fronte ad una mole di lavoro sempre crescente. Il loro costo di gestione, inoltre, era molto elevato (si rendevano necessarie montagne di carta e continue manutenzioni) ed avevano una capacità (ed una velocità) di visualizzazione abbastanza limitata rispetto alle esigenze dell'utenza.
Fu così che si pensò di affiancare alle pur necessarie stampanti delle unità di output più adeguate e flessibili all'uso di quanto non fossero quelle utilizzate fino a quel momento.
La scelta, come certamente avrai già immaginato, cadde sugli schermi video.
Essi rispondevano a tutti i requisiti richiesti: erano compatti, affidabili, economici (scarsa manutenzione e ridottissime spese di gestione), veloci.
Da allora, anno dopo anno, il loro utilizzo è diventato sempre più esteso ed intenso: ormai ai giorni nostri è impossibile riuscire a trovare un qualsiasi calcolatore che non sia provvisto di unità video.
Nei moderni personal le unità video che di solito vengono impiegate sono principalmente di due tipi: televisori e monitor.
Tra un televisore ed un monitor non esiste, fisicamente e sostanzialmente, una grande differenza. Un monitor, infatti, non è

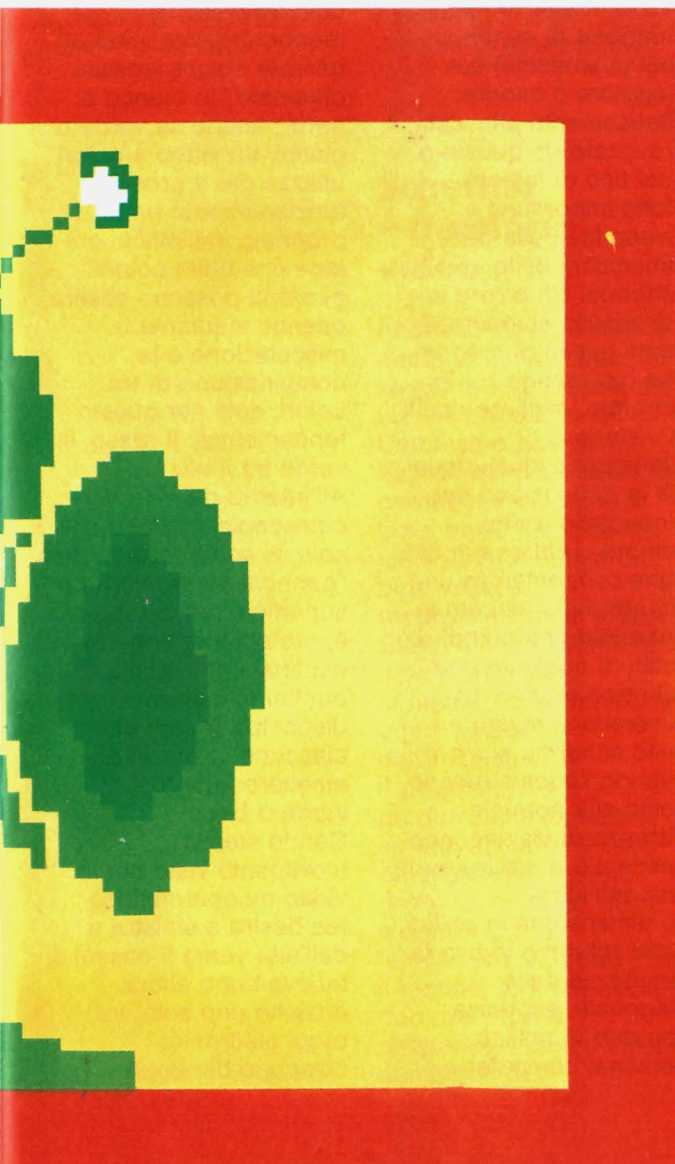
altro che un televisore di ottima qualità al quale sono stati asportati tutti i circuiti adatti per la ricezione dei segnali attraverso l'antenna.
La qualità dell'immagine è ovviamente superiore a quella ottenibile da un normale televisore. Per un uso non professionale può essere comunque non necessario (se non inutile) ricorrere all'acquisto di un monitor: anche il televisore di casa è infatti in grado di svolgere egregiamente il lavoro di visualizzazione, ad un prezzo sicuramente più contenuto.
Il modo in cui vengono prodotte le scritte sul display video del tuo VIC 20 è abbastanza semplice: esiste infatti una zona della memoria RAM nella quale vengono depositati - sotto forma di codice ASCII - tutti i caratteri che devono essere presentati sullo schermo.
A tale zona fa riferimento un apposito circuito, interno al VIC 20, che "interfaccia" il calcolatore con l'unità video. Esso preleva cioè tutti i caratteri presenti nelle varie locazioni della memoria video e li invia al circuito del televisore, sotto forma di impulsi elettrici compatibili con il

sistema o lo standard televisivo.
Il televisore (od il monitor video) produce quindi una immagine visibile partendo da questi segnali elettrici.
Il modo in cui ciò avviene costituisce una delle più interessanti ed utili applicazioni della fisica elettronica e vede come componente principale un dispositivo chiamato cinescopio.
Un cinescopio (o tubo catodico) è un tubo a vuoto, cioè un contenitore di vetro nel quale è stato fatto il vuoto.
All'interno del tubo si

trova un "cannone elettronico" (basato su un filamento riscaldato dalla corrente che lo attraversa, come nelle lampadine) che produce un fascio (o pennello) di elettroni molto sottile. Gli elettroni godono di una singolare proprietà: quando colpiscono particolari sostanze fluorescenti - chiamate anche fosfori - fanno generare a queste sostanze una luminescenza, la cui durata può andare da alcuni millisecondi (millesimi di secondo) ad alcuni secondi, in funzione del tipo di fosforo e dell'intensità del raggio elettronico. L'immagine viene ricostruita proprio sfruttando questo fatto: il fascio di elettroni, "sparato" dal cannone e comandato da opportuni campi elettrici e magnetici, spostandosi da destra a sinistra e dall'alto verso il basso applica maggiore o minore intensità ai singoli punti di uno schermo che è stato ricoperto da un sottile strato di fosfori, provocandone una maggiore o minore luminosità.

La bassa risoluzione utilizza esclusivamente i caratteri (normali o grafici) presenti sulla tastiera. Le immagini così ottenute non risultato molto dettagliate.

L'immagine viene pertanto ricostruita punto per punto dal pennello elettronico del tubo catodico alla stessa maniera in cui l'occhio di un uomo legge la pagina stampata di un giornale o di un libro. La velocità con cui tale pennello attraversa l'intero schermo è però talmente elevata da non poter essere minimamente percepita dall'osservatore umano: lo standard televisivo europeo prevede infatti che l'intero schermo, suddiviso in 625 righe orizzontali (mentre nello standard americano le

linee sono 525) sia percorso completamente ogni cinquantesimo di secondo. Il fenomeno della persistenza dell'immagine sulla retina fornisce quindi l'illusione di una immagine completa e simultanea.

All'interno del televisore (o del monitor) vi sono, come accennato, anche dei circuiti destinati a comandare i movimenti del raggio elettronico sia in senso orizzontale che verticale.
Perché l'immagine sia visibile è però necessario che questi dispositivi lavorino simultaneamente.
Nel segnale video sono allora compresi, oltre alle informazioni riguardo all'intensità di ogni singolo punto dello schermo, anche appositi segnali destinati a coordinare il movimento del pennello elettronico. Sono i segnali di sincronismo.
Nel caso della trasmissione televisiva tutte queste informazioni provengono da una telecamera; per i computer, invece, esiste un apposito circuito che, proprio come una telecamera, "legge"

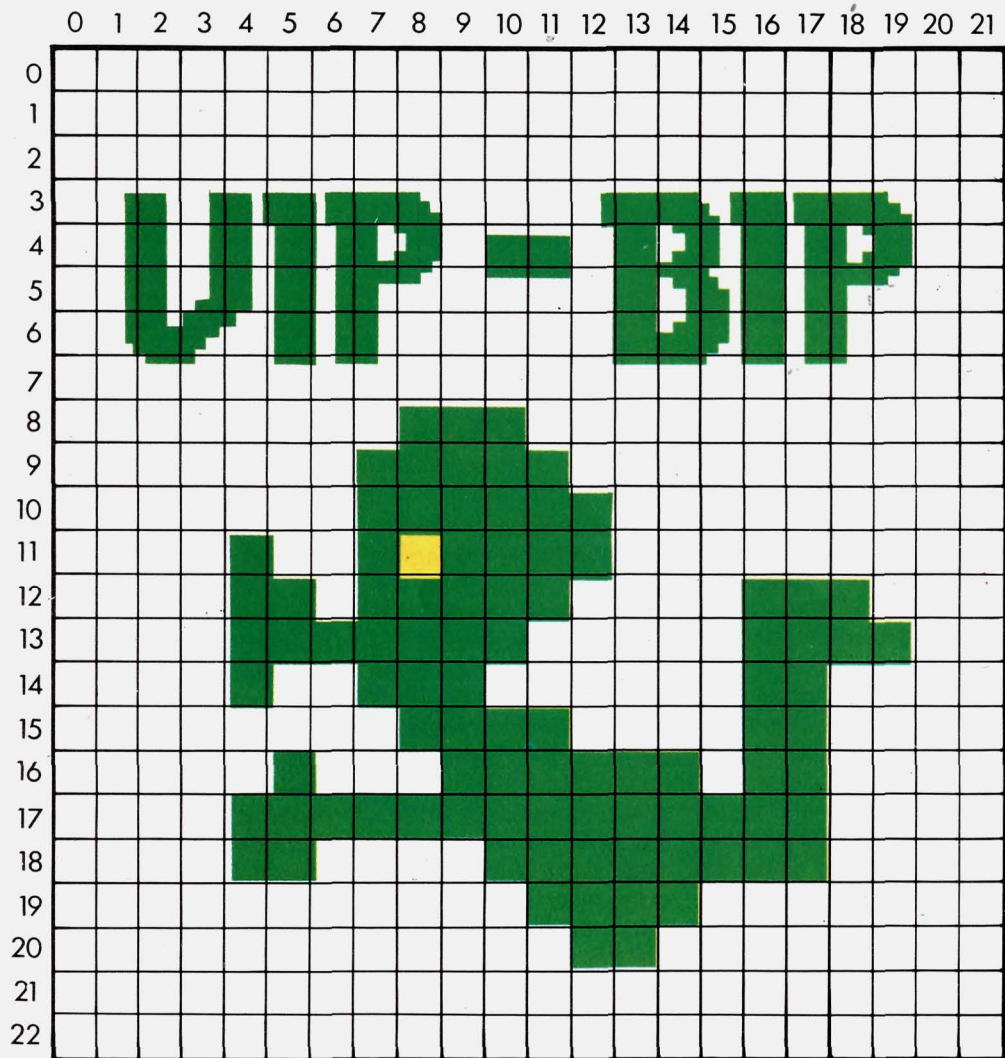
**Con l'alta risoluzione puoi indirizzare il singolo pixel determinandone o meno l'accensione.
Le immagini così ottenute risultano molto dettagliate.**

l'immagine da visualizzare nelle varie locazioni riservate alla memoria video.
Il tipo di fosforo applicato sulla superficie dello schermo determina anche il colore del punto di collisione tra pennello elettronico e fosforo stesso.
Esistono in commercio diversi tipi di monitor: a fosfori verdi, gialli, ambra, ecc.. La scelta di un colore rispetto ad un altro è normalmente una questione di gusti e preferenze personali, anche se ultimamente giungono di continuo voci (e smentite) sul maggiore o minore affaticamento alla vista provocato da questo o quel tipo di fosforo.
Molto importante è invece la scelta delle dimensioni dello schermo. Un errore in cui spesso si incorre è infatti quello di credere che più grande sia lo schermo, migliore risulti la visione.
L'immagine, qualunque sia la dimensione del cinescopio, viene sempre suddivisa in 625 righe orizzontali: in uno schermo più piccolo le righe saranno quindi più sottili di quelle in uno schermo grande. La dimensione migliore dello schermo si avrà quando l'occhio umano, posto alla normale distanza di visione, non riesce più a distinguerne una dall'altra.
La dimensione in pollici dello schermo indica la lunghezza della diagonale, espressa appunto in pollici: personal computer e terminali video hanno solitamente schermi da 9" o 12" (9 o 12 pollici).
Un discorso a parte meritano gli schermi a colori.
Fermi restando i principi della televisione monocromatica, cioé ad un solo colore (spesso chiamato "in bianco e nero", anche se verde o giallo), un video a colori utilizza per il proprio funzionamento una proprietà dell'ottica, e cioé che tutti i colori esistenti possono essere ottenuti mediante la miscelazione e la combinazione di tre colori, detti per questo fondamentali: il rosso, il verde ed il blu.
All'interno del cinescopio, anziché uno solo, vi sono pertanto tre "cannoni elettronici". La superficie del cinescopio è interamente coperta da centinaia di migliaia di puntini di fosforo disposti a gruppi di tre, ciascuno in grado di emettere luce rossa, verde o blu.
Con lo stesso movimento visto per il video monocromatico (da destra a sinistra e dall'alto verso il basso) si muoveranno allora, anziché uno solo, tre raggi elettronici, ciascuno dei quali -

passando attraverso una maschera metallica opportunamente forata - è in grado di colpire soltanto i fosfori di un determinato colore.
Sulla superficie dello schermo si ottengono quindi tre immagini coincidenti, rispettivamente di colore rosso, verde e blu che combinandosi tra loro formano per l'occhio umano un'unica immagine a colori.
Questa tecnica si chiama sintesi additiva.

Schermo e memoria video

Abbiamo dunque visto il modo in cui le immagini vengono visualizzate sul tuo schermo televisivo.
Cerchiamo adesso di approfondire la relazione esistente tra ciò che si trova nella memoria del tuo VIC 20 e quello che puoi vedere rappresentato sul video.
Esiste infatti una stretta dipendenza tra contenuto della memoria e formazione dell'immagine.
Il calcolatore, per produrre una immagine su uno schermo video, deve generare (come detto) un segnale simile a quello di una telecamera, in modo che il monitor (o il televisore) non noti alcuna differenza. I costruttori di computer, di conseguenza, sono stati costretti a ricorrere a stratagemmi per riuscire ad "ingannare" il video.
Ti ricordi quando, un paio di lezioni fa, parlammo della mappa della memoria del VIC 20?
Bene, ora è arrivato il momento di rinfrescare l'argomento.
La memoria di un elaboratore non è a completa disposizione dell'utente: esistono infatti alcune zone (o aree) che non sono direttamente e liberamente utilizzabili per inserire dati ed istruzioni, ma che assolvono invece compiti, diciamo così, di ausilio e supporto.
A tali porzioni della memoria vengono quindi affidati, in sede di progetto e costruzione, incarichi non propriamente di "elaborazione", ma non per questo meno importanti ai fini del buon funzionamento del calcolatore. Così, alcune locazioni sono state dedicate a contenere i programmi, altre l'interprete BASIC, altre ancora - e sono quelle che ci interessano oggi - i caratteri da visualizzare sullo schermo.
L'area della memoria di

HARDWARE

cui ci vogliamo occupare prende il nome di memoria video. Il suo scopo è quello di contenere tutte le informazioni necessarie a costruire una immagine sul display televisivo.
Secondo te, sarà una memoria RAM od una memoria ROM? Non dovresti avere molti dubbi.

Per forza di cose, infatti, dal momento che vogliamo poter modificare il contenuto dello schermo, la memoria video deve appartenere alla zona di memoria RAM. In essa devono quindi avvenire cambiamenti tutte le volte in cui, per esempio, esegui una istruzione PRINT o INPUT, oppure (a parte casi particolari) batti qualcosa sulla tastiera che deve essere visualizzato sullo schermo.
Da sola, però, la memoria video non può fare molto: è infatti necessario che un apposito circuito video, chiamato di refresh (cioè "rinfresco") dell'immagine, legga - esattamente come una telecamera - tutta la memoria, carattere per carattere.
A quel punto il circuito video, venuto a conoscenza dei codici dei caratteri da visualizzare, consulta una particolare memoria ROM (chiamata anche generatore di caratteri) dalla quale preleva la "descrizione" grafica del carattere stesso. Come risultato finale si ottiene quindi un segnale video indicante se ciascun punto dello schermo deve essere acceso o spento.
Tutto ciò che compare sul video viene infatti rappresentato sotto forma di particolari combinazioni di puntini luminosi, chiamati pixel (abbreviazione dell'inglese picture element).
Ciascun puntino, potendo essere solo acceso o spento, è pertanto descrivibile da un singolo bit.
Senza addentrarci ulteriormente in dettagli tecnici, ti basti sapere che il circuito di refresh "prende" la combinazione di bit contenente la descrizione del carattere e la mette nella corrispondente casella dello schermo, accendendo o spegnendo i puntini come indicato precedentemente.
In totale i pixel luminosi disponibili sul tuo VIC 20 sono 32.384. Ad essi corrisponde nel complesso uno schermo composto da:
23 righe e 22 colonne, che formano in totale 23*22 = 506 caratteri.

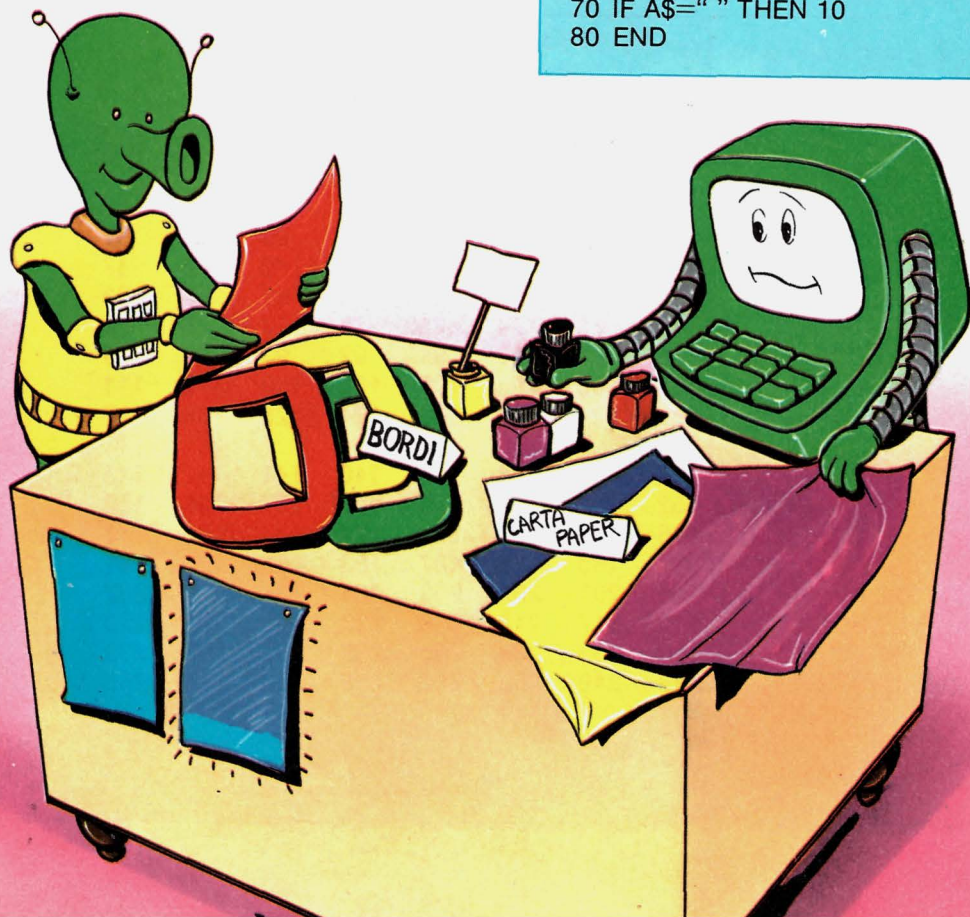
Gli attributi e i colori

Lo schermo è quindi idealmente (ma anche fisicamente) divisibile in 506 (23 righe di 22 caratteri) posizioni dove i caratteri possono essere stampati. Ognuno di essi è rappresentato quindi da un quadrato di punti con dimensione 8 * 8.
Ciascuno di questi caratteri è inoltre rappresentabile in vari modi, per esempio: in

colore in reverse e in positivo.
Puoi cioè stabilirne gli attributi.
Così, quando stampi qualcosa sul video non fai altro che modificare la combinazione di alcuni degli attributi assegnati in precedenza a quella posizione.
Normalmente, cioè quando non imponi particolari specifiche - per esempio, riguardo il colore -, l'unico attributo che subisce modifiche è soltanto quello che riguarda l'accensione o lo spegnimento dei pixel che devono comporre il carattere.
Tutti gli altri attributi restano inalterati.
Esiste comunque la possibilità di variare a piacimento anche gli altri attributi.
Sfortunatamente però, il tuo VIC 20 non dispone di comandi specificamente destinati a questo scopo. È allora necessario ricorrere al magico comando POKE che, alterando il contenuto di particolari locazioni, ti consente di ottenere lo stesso i risultati desiderati.
Parliamo per prima cosa dei colori disponibili sul tuo VIC 20. Come puoi vedere nella sottostante tabella, a ciascuno di essi è associato un numero (compreso tra 0 e 7 ed indicante appunto il colore) ed un valore ASCII. Quest'ultimo indica il codice del particolare carattere di controllo - del colore, appunto - assegnato a ciascuno dei colori e disponibile direttamente sulla tastiera (tramite la pressione contemporanea del tasto CTRL e di uno dei tasti numerici compresi tra 1 e 8).
Sul manuale del tuo calcolatore troverai comunque al riguardo notizie ed informazioni più dettagliate.
In un televisore in bianco e nero ciascuno di questi colori corrisponde ad una diversa gradazione di grigio, più o meno intensa, a seconda della

COLORE	NUMERO COLORE	CHR$
NERO	0	144
BIANCO	1	5
ROSSO	2	28
CIANO	3	159
PORPORA	4	156
VERDE	5	30
BLU	6	31
GIALLO	7	158

tonalità del colore.
La locazione che contiene il valore del colore dei caratteri è la 646. Modificando il valore in essa contenuto è quindi possibile cambiare il colore di tutto ciò che verrà visualizzato da lì in avanti.
Il seguente programma ti aiuterà a chiarire le idee (per terminare premi un tasto qualsiasi):

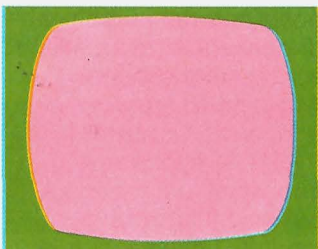
```
10 FOR I=0 TO 7
20 POKE 646,I
30 PRINT "PROVA DI STAMPA"
40 FOR J=0 TO 50:NEXT J
50 NEXT I
60 GET A$
70 IF A$=" " THEN 10
80 END
```

Al momento dell'accensione nella celletta 646 è presente il valore 6; il colore del cursore è pertanto blu. La locazione 36879 contiene invece il valore dei colori del bordo e dello sfondo.
Leggi nella tabella i valori che devi "POKARE" nella locazione 36879 per ottenere le combinazioni

Combinazione dei colori del margine e dello schermo

	Margine							
Schermo	BLK (Nero)	WHT (Bianco)	RED (Rosso)	CYAN (Blu-Verde)	PUR (Porp.)	GRN (Verde)	BLU (Blu)	YEL (Giallo)
NERO	8	9	10	11	12	13	14	15
BIANCO	24	25	26	27	28	29	30	31
ROSSO	40	41	42	43	44	45	46	47
BLU-VERDE	56	57	58	59	60	61	62	63
PORPORA	72	73	74	75	76	77	78	79
VERDE	88	89	90	91	92	93	94	95
BLU	104	105	106	107	108	109	110	111
GIALLO	120	121	122	123	124	125	126	127
ARANCIO	136	137	138	139	140	141	142	143
ARAN. CHIARO	152	153	154	155	156	157	158	159
ROSA	168	169	170	171	172	173	174	175
AZZURRO CHI.	184	185	186	187	188	189	190	191
PORPORA CHI.	200	201	202	203	204	205	206	207
VERDE CHI.	216	217	218	219	220	221	222	223
AZZURRO	232	233	234	235	236	237	238	239
GIALLO CHI.	248	249	250	251	252	253	254	255

di colori dello sfondo e del bordo che desideri. Ad esempio se introduci:

```
POKE 36879,173
```

otterrai uno sfondo rosa con bordo verde.

Un'ultima cosa. Se il codice dello sfondo e quello dei caratteri coincidono - cioè se sia lo sfondo che i caratteri hanno lo stesso colore - accade un fatto facilmente immaginabile: non si legge niente. I caratteri vengono infatti scritti con lo stesso colore dello sfondo, proprio come se un pennarello bianco scrivesse su un foglio di carta bianco. Attenzione quindi!

Riguardo alla possibilità di rappresentare i caratteri in campo inverso, il tuo VIC 20 dispone di un apposito carattere di controllo (codificato in ASCII). Tale carattere è ottenibile mediante la pressione contemporanea dei tasti CTRL e 9 e corrisponde alla stampa di CHR$(18). Per tornare al modo normale si devono premere contemporaneamente i tasti CTRL e 0; questa azione corrisponde alla stampa di CHR$ (146). Vediamo un esempio:

```
10 FOR I=0 TO 10
20 PRINT CHR$ (18); "PROVA STAMPA INVERSA"
30 PRINT CHR$ (146); "PROVA STAMPA NORMALE"
40 NEXT I
```

Funzioni di controllo della stampa su video

Nell'ambito delle istruzioni di visualizzazione previste dal BASIC, particolare importanza rivestono tutti *i comandi che* controllano e modificano, a piacere del programmatore, la posizione del cursore e - di conseguenza - delle scritte sullo schermo.
Abbiamo finora adoperato l'istruzione PRINT in numerose occasioni, utilizzandola per visualizzare tutti i risultati, i messaggi e le scritte che di volta in volta ci sono stati *convenienti o necessari.*
Ciò che però ancora ci manca è la capacità di controllare completamente questa istruzione, consentendoci per esempio di ottenere in uscita i risultati disposti in una certa posizione dello schermo, oppure ordinati ed incolonnati in un formato non necessariamente impostoci dal nostro *calcolatore.*
Detto in una parola (peraltro molto usata nel ramo dell'informatica), vogliamo saper formattare le scritte sullo schermo.
Questo è pertanto l'obiettivo delle istruzioni che oggi ci proponiamo di imparare.
La cosa importante da sottolineare è che tutte queste istruzioni non indicano al calcolatore *cosa stampare, ma* soltanto DOVE stampare.
Capito la differenza?
Vediamole adesso una per una, insieme a qualche esempio esplicativo.

SPC

La funzione SPC () è utilizzata all'interno di istruzioni PRINT per *scrivere un certo* numero di spazi.
L'argomento numerico assegnato alla funzione specifica il numero di spazi da scrivere.
SPC() permette quindi di spostare su una linea il cursore di quante posizioni si vuole.
L'argomento che fornirai dovrà essere un valore numerico intero compreso tra 0 e 255 (se vi saranno anche cifre *decimali,* queste verranno eliminate automaticamente); in caso contrario il tuo VIC 20 ti invierà il messaggio di errore:

```
ILLEGAL QUANTITY
ERROR
```

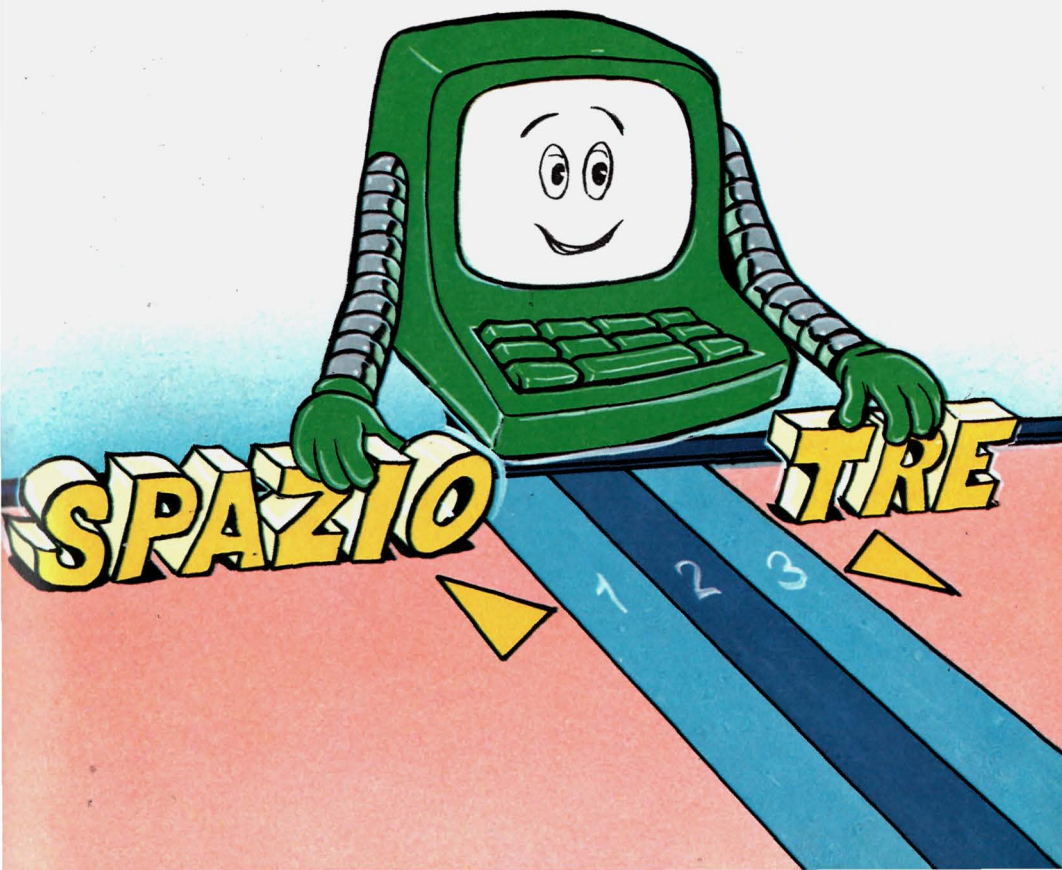
Esempi

```
PRINT "ROSSO"; SPC (4); "DI"; SPC (3); "SERA"
```

Questa funzione interpone 4 spazi tra le parole ROSSO e DI, mentre tra le parole DI e SERA inserisce 3 spazi.

Nota come dopo gli SPC compaia il punto e virgola. Se infatti vi fosse stata una virgola, avremmo avuto anche l'effetto di tale virgola, ottenendo in uscita le parole ancora più distanziate.

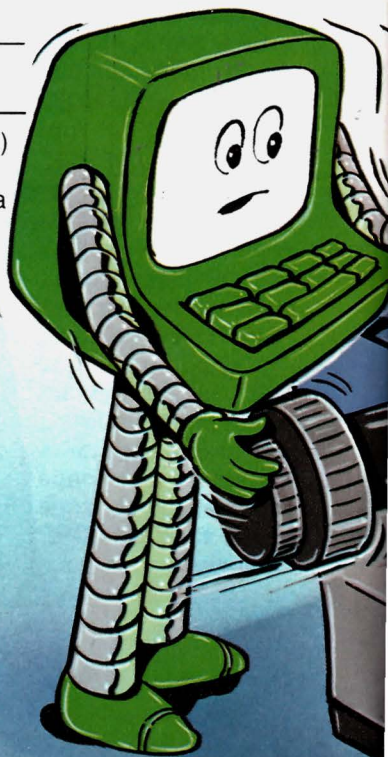
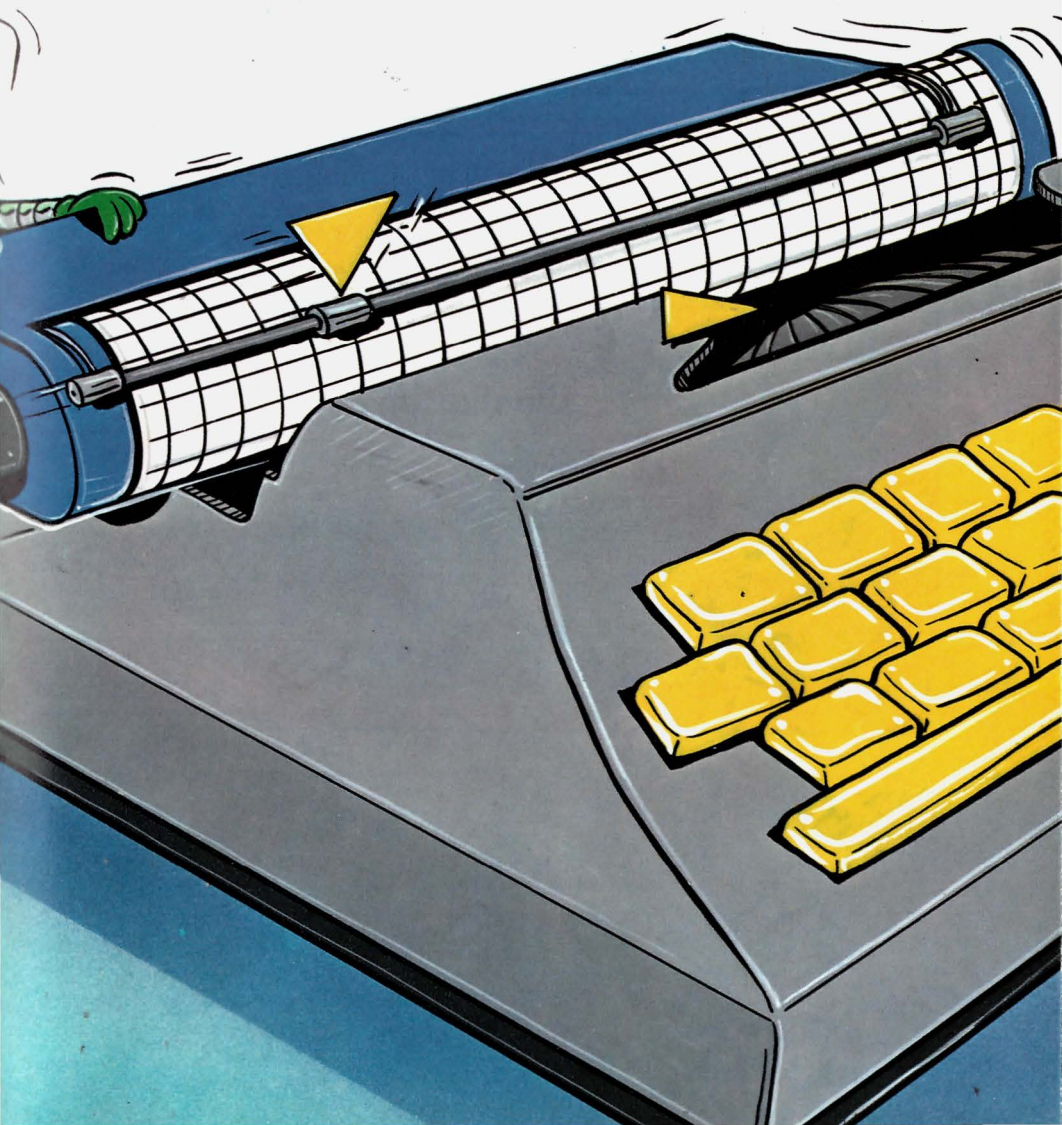
```
10 FOR I=0 TO 18
20 PRINT SPC(I); "CIAO"
30 NEXT
```

Questo breve programma stampa invece una serie di saluti, spostandoli man mano sullo schermo.

Sintassi della funzione

SPC (espressione)

dove espressione deve essere un valore numerico intero compreso tra 0 e 255

TAB

Anche la funzione TAB() si usa solo all'interno delle PRINT: essa opera infatti come la tabulazione di una normale macchina da scrivere.
TAB() si distingue da SPC(), in quanto sposta il cursore su una certa colonna in modo assoluto (cominciando cioè a contare dalla colonna numero 0), mentre SPC() lo sposta

di un certo numero di colonne in modo relativo (contando cioè gli spazi partendo dall'attuale posizione di stampa).
La funzione TAB() viene normalmente utilizzata per allineare le stampe in colonne verticali, incolonnandole in punti prestabiliti.
Supponi infatti di dover visualizzare alcuni dati sullo schermo del tuo video, allineati in un certo ordine. Grazie a TAB() puoi evitare i noiosi (e talvolta complicati) calcoli per incolonnare esattamente tutti gli elementi.

Esempi

```
PRINT NOME$; TAB(13);
COGNOME$
```

Con questa istruzione, indipendentemente dalla lunghezza della stringa contenuta in NOME$, si avrà la stampa delle due variabili con questa disposizione: la prima partendo dalla colonna 0, la seconda dalla colonna 13.

```
10 PRINT TAB(1); "NUMERO"; TAB(8);
   "QUADRATO"
20 FOR I=1 TO 15
30 PRINT TAB(1);I;TAB(8);I↑2
40 NEXT
```

Il programma qui a fianco mostra una semplice applicazione di TAB: stampa infatti, allineati per colonna, i primi 15 numeri con i rispettivi quadrati:

Sintassi della funzione

TAB (espressione)

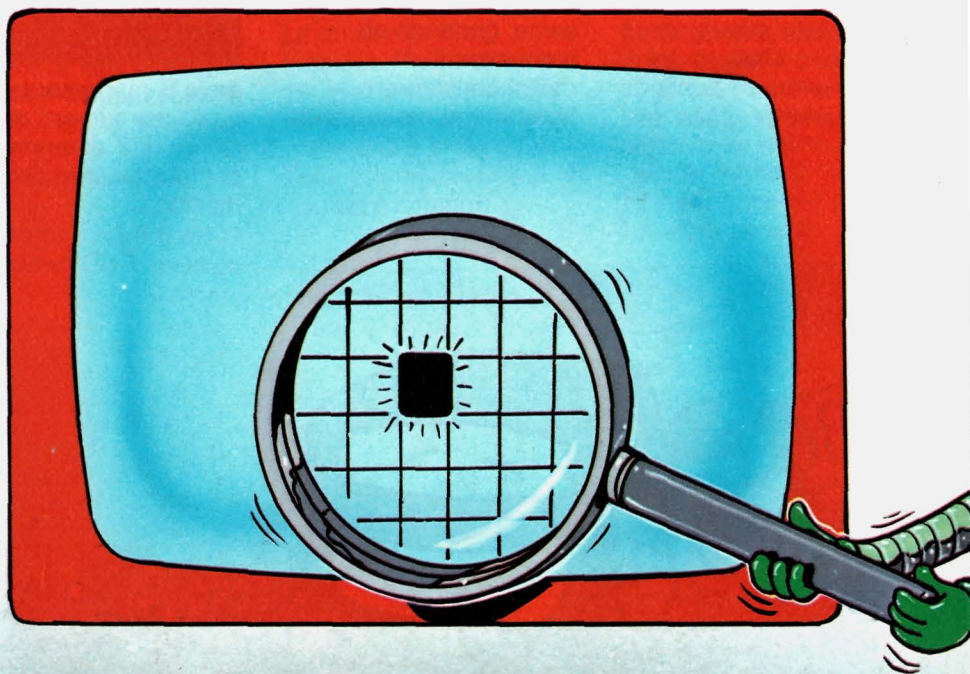
dove espressione è un valore numerico compreso tra 0 e 255.

POS

La funzione POS() restituisce la posizione attuale del cursore in senso orizzontale, cioè la colonna sulla quale verrà stampato il carattere successivo. Essa considera però una linea video come formata da due righe, 40+40 caratteri; dà

quindi come risultato un numero compreso tra 0 e 79.
L'argomento della funzione (strano, ma vero!) non ha alcuna importanza; deve però essere sintatticamente corretto (normalmente si utilizza, lo zero).
Ad esempio il seguente breve programma,

```
10 PRINT SPC(5);"CANE";SPC(5)
20 PRINT POS(0)
```

provocherà la visualizzazione della parola CANE preceduta e seguita da 5 spazi (che naturalmente risultano invisibili all'occhio umano).
La riga 20 stamperà *quindi* la posizione attuale del cursore, cioè 5+4+5 = 14

Sintassi della funzione

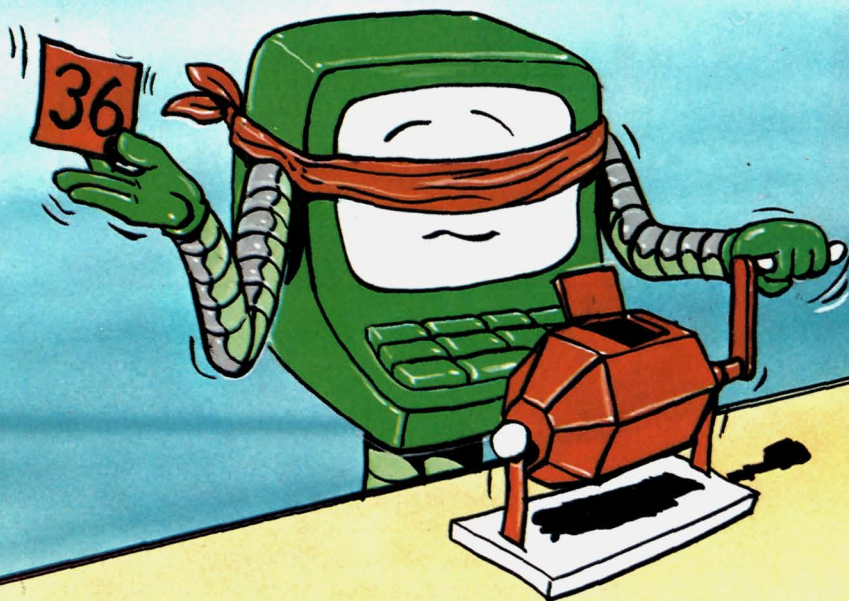
POS (argomento)

dove argomento è un valore numerico puramente formale, senza cioè alcuna specifica funzione.

RND

Non è raro trovare casi in cui occorra, nell'ambito di un programma, disporre di *numeri casuali*. Potresti, ad esempio, aver bisogno di un algoritmo che simuli l'estrazione a sorte di un numero, o il lancio dei dadi o ancora l'uscita delle carte da gioco dal mazzo.
La funzione RND ti risolve egregiamente questo tipo di problemi: crea, infatti, dei numeri casuali (random in inglese) compresi tra 0 e 1 (0 < = R < 1).
Il tuo VIC 20 produce una sequenza di numeri casuali eseguendo dei calcoli su di un numero di partenza chiamato seme, generato al momento dall'accensione.
Poiché i numeri generati da RND sono il risultato di una complessa serie di calcoli, è più corretto parlare di numeri pseudocasuali.
L'argomento di RND controlla il seme della funzione e determina il punto di partenza della serie dei numeri casuali.
Il formato della funzione è:

```
RND (S)
```

dove S rappresenta un qualsiasi numero reale. Il valore di S determina il comportamento della funzione:
1) Se S è un numero positivo, RND genererà

una imprevedibile serie di numeri.
2) Con S = 0 si otterrà ugualmente una serie di numeri casuali, generati però con calcoli diversi, basati in questo caso sul valore dell'orologio interno del tuo VIC 20.
3) Per un valore di S negativo, RND ritornerà sempre la stessa serie di numeri pseudocasuali: in altre parole, puoi ottenere la stessa serie di numeri semplicemente richiamando la funzione RND e fornendole lo stesso argomento negativo.

Ad esempio, il programma che segue produrrà sempre la stessa serie di numeri random indipendentemente da quando darai il RUN e da quante volte lo farai:

```
10 PRINT RND (– 1)
20 FOR I = 1 TO 5
30 PRINT RND (1)
40 NEXT I
```

La chiamata alla funzione RND nella linea 10, grazie all'argomento negativo, determina una prefissata e prevedibile serie di numeri. Come risultato, i numeri random ottenuti all'interno del LOOP del ciclo FOR ... NEXT saranno sempre gli stessi 5.
Se cancelli invece la linea 10, il punto di partenza della serie sarà di volta in volta diverso, così come i 5 numeri generati.
Poiché la gamma dei numeri compresi tra 0 ed 1 non è adeguata alla maggior parte delle applicazioni, usa la seguente formula per stabilire tu stesso il "range" entro il quale dovranno essere prodotti i numeri casuali:

```
LET R = INT ((LS – LI + 1) * RND (1)) + LI
```

Dove R sta per numero Random, LS è il Limite Superiore del "range" ed *LI il Limite Inferiore.*
Per numeri Random interi compresi tra 1 e LS (variabile numerica da definire precedentemente), applicando la formula avrai:

```
LET R = INT (LS * RND (1)) + 1
```

Sintassi della funzione

RND (Espressione Numerica)

Il modo virgolette

Abbiamo già visto in alcune occasioni come la pressione contemporanea di alcune coppie di tasti provochi l'esecuzione immediata di particolari azioni od operazioni: per esempio, cancellazione del video, modifica dei colori, spostamento del cursore.
A ciascuna di queste azioni corrisponde un particolare codice ASCII.
All'interno dei programmi

esiste pertanto la completa facoltà di utilizzarle liberamente, specificandone il relativo codice. Così, se volessimo cancellare lo schermo e riportare il cursore in alto a sinistra, potremmo impartire

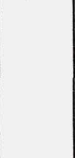
```
PRINT CHR$ (147)
```

È comunque possibile incorporare in una stringa i caratteri speciali che corrispondono a questi originali (ma indispensabili) comandi. Quando, nel corso del programma, la stringa verrà stampata, si potrà così ottenere lo stesso effetto del codice di controllo assegnato attraverso la funzione CHR$.
Nel programma che segue puoi vedere tutte queste cose. Esso opera così:

- pone nelle stringhe A$, B$, C$, D$ i caratteri che si generano, premendo contemporaneamente CTRL ed i tasti da 1 a 4 (e che controllano 4 degli 8 colori disponibili sul tuo VIC 20);
- stampa tali stringhe una dopo l'altra, alternandole con un ciclo di attesa, in modo da lasciare il tempo di renderti conto dell'effetto di ciascuna di esse

```
10 A$="■":B$="◘":C$="◙":D$= "◣" : E$ = "CIAO"
20 PRINT A$; E$ : FOR I=0 TO 100 : NEXT
30 PRINT B$; E$ : FOR I=0 TO 100 : NEXT
40 PRINT C$; E$ : FOR I=0 TO 100 : NEXT
50 PRINT D$; E$ : FOR I=0 TO 100 : NEXT
60 GOTO 20
```

Questo modo di inserire i caratteri di controllo nelle stringhe è una peculiarità tipica del tuo VIC 20. Si è soliti chiamarlo "modo virgolette", appunto

perché è proprio all'interno delle virgolette che trova ragione di esistere.
Nella tabella che segue illustriamo il significato dei caratteri che permettono di muoversi sul video e di controllare il colore.
Nella prima colonna appaiono i tasti da premere, nella seconda il carattere visualizzato sul video in modo virgolette, nella terza il loro effetto una volta stampati.

CLR/HOME	S	Manda il cursore nell'angolo in alto a sinistra del video, senza cancel- lare il suo contenuto. Corrisponde a CHR$ (19).
SHIFT + CLR/HOME		Pulisce il video e manda il cursore nell'angolo in alto a sinistra. Corrisponde a CHR$ (147).
CRSR/frecce-verticali	Q	Sposta il cursore di una posizione verso il basso. Corrisponde a CHR$ (17).
SHIFT + CRSR/frecce-verticali		Sposta il cursore di una posizione verso l'alto. Corrisponde a CHR$ (145).
CRSR/frecce-orizzontali	]	Sposta il cursore di una posizione verso destra. Corrisponde a CHR$ (29).
SHIFT + CRSR/frecce-orizzontali		Sposta il cursore di una posizione verso sinistra. Corrisponde a CHR$ (157).
CTRL + 9	R	Permette di visualizzare i caratteri che seguono in campo inverso. Corrisponde a CHR$ (18).
CTRL + 0		Pone fine alla sequenza dei caratteri in campo inverso. Corrisponde a CHR$ (146).

CTRL + 1		Nero
CTRL + 2		Bianco
CTRL + 3		Rosso
CTRL + 4		Ciano
CTRL + 5		Porpora
CTRL + 6		Verde
CTRL + 7		Blu
CTRL + 8		Giallo

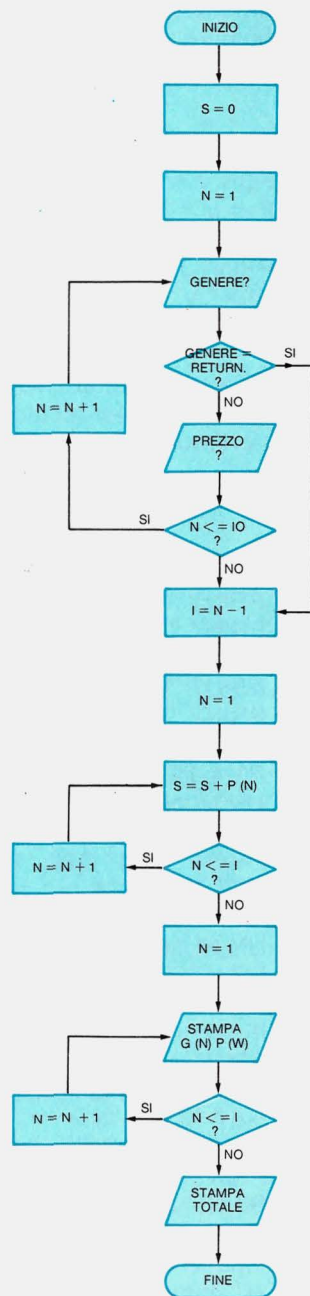
Per modificare i dati presenti sullo schermo video si deve portare il cursore nella posizione desiderata e, poi, o riscrivere sopra o inserire o cancellare i caratteri. I tasti per inserire o cancellare sono:

INST/DEL		Cancella il carattere a sinistra del cursore e sposta il cursore e tutto quello che lo segue verso sinistra di una posizione. Corrisponde a CHR$ (20).
SHIFT + INST/DEL		Crea uno spazio a sinistra del cursore, spostando il cursore e tutto quello che lo segue di una posizione verso destra. Corrisponde a CHR$ (148).

Lista della spesa

Il programma che segue somma le voci di una spesa giornaliera. Puoi introdurre fino a dieci generi con i relativi importi di acquisto. Nota come, per ottenere una corretta visualizzazione dei dati, si sia fatto ricorso alla funzione TAB.

SPESA DEL GIORNO
ALIMENTARI 32.000
CINEMA 3.000
RIVISTE 6.000
TELEFONO 105.000
TOTALE 146.000

```
10 FOR N = 1 TO 10
15 INPUT "GENERE"; G$ (N)
20 IFG$ (N) = " " THEN GO TO 35
25 INPUT "PREZZO"; P (N)
30 NEXT N
35 LET I = N - 1 : LET S = 0
40 FOR N = 1 TO I
45 LETS = S + P (N)
50 NEXT N
55 PRINT "♥ Q ] ] LISTA DELLA SPESA Q"
60 FOR N = 1 TO I
65 PRINT G$ (N); TAB (12); P (N)
70 NEXT N
75 PRINT "Q TOTALE"; TAB (12); S
80 END
```

Usando la funzione TAB, ricerca un nuovo metodo per stampare la tabellina di un numero, in modo che le unità, le decine ecc. dei vari prodotti risultino perfettamente incolonnate. Ecco, per aiutarti, un esempio con quella del 7.

```
10 C = 10 : S = C : REM LE UNITA' VANNO STAMPATE ALLA COLONNA 10
20 PRINT "■ "
30 FOR V = 1 TO 10
40 P = V * 7
50 IFP > 9 THEN S = C - 1
60 PRINT TAB (S); P
70 NEXT V
80 END
```

Prova a sostituire la linea 30 con : 30 FOR V = 1 TO 20.
Suggerimento: per incolonnare le centinaia, introduci tra la linea 50 e la 60 un ulteriore controllo.

Gioca a dadi con il tuo VIC 20

```
 10 PRINT "■ "
 20 PRINT "[Q][]] []][]] DADOMATTO [Q][Q]"
 30 INPUT "GIOCATORE 1 =  "; G1$
 40 INPUT "[Q] [Q] GIOCATORE 2 =  "; G2$
 50 PRINT "[Q] [Q] G1$;" PREMI UN TASTO"
 60 GET T$ : IF T$ = " " THEN 60
 70 R1 = INT (6 * RND (1)) + 1 : REM CON UN DADO PUO' USCIRE SOLTANTO
    UN NUMERO TRA 1 E 6
 80 PRINT TAB (10); R1
 90 PRINT "[Q][Q]" G2$; "PREMI UN TASTO"
100 GET T$ : IF T$ = " " THEN 100
110 R2 = INT (6 * RND (1)) + 1 : REM CON UN DADO PUO' USCIRE SOLTANTO
    UN NUMERO TRA 1 E 6
120 PRINT TAB (10); R2
130 IF R1 = R2 THEN PRINT "[Q] [Q] PARTITA PARI" : GOTO 200
140 IF R1 > R2 THEN PRINT "[Q] [Q] VINCE"; G1$ : GOTO 200
150 PRINT "[Q][Q] VINCE"; G2$
200 PRINT "[Q][]] []] ANCORA ? S/N"
210 GET T$ : IF T$ = " " THEN 210
220 IF T$ = "N" THEN END
230 PRINT " ■ " : GOTO 50 : REM CON UN TASTO DIVERSO DA "N" IL GIOCO
    RIPRENDE
```

Modifica le linee 70 e 110 in modo da simulare il lancio di 2 dadi: pensa a quali numeri, minimo e massimo possono uscire.